# CHAPTER No. 1 INTRODUCTION

## 1.1 Introduction to Project

As we are at the end of the BS Degree, we have to submit the "Final Year Project" to our college. The project that we worked on is about the Tutor Finder where a user finds a qualified tutor. Our project is basically a website by the name of Tutor Finder.

The website has admin and tutor panel with user interface. The user can visit the site and see the available qualified tutor and he/she can hire a tutor. The admin can manage the whole system including all tutor and their details.

## 1.2 Objective

- Connect Students with Tutors.
- Accessibility
- Matching:
- Quality Assurance:
- Scheduling and Management
- Privacy and Security
- Education and Support

## 1.3 Purpose

The purpose of the Tutor Finder project is to bridge the educational gap by providing a user-friendly and accessible platform that connects students seeking academic support with qualified tutors. In a rapidly evolving educational landscape, the project aims to empower learners of all ages and backgrounds by facilitating personalized, one-on-one or group tutoring sessions across various subjects and skill levels. By harnessing technology, the project not only makes tutoring services readily available to a broad spectrum of learners but also ensures the quality and transparency of these services through rigorous tutor vetting processes, user reviews, and continuous improvement mechanisms. Through this project, we aspire to democratize education, break down geographical barriers, and foster a sense of community among students and tutors, ultimately enabling individuals to unlock their full learning potential, achieve their academic goals, and embark on a journey of lifelong learning and personal growth.

## 1.4 Scope of Project

The scope of the Tutor Finder project encompasses the development and deployment of a comprehensive online platform or mobile application that facilitates the seamless matching of students with suitable tutors, enabling them to access tutoring services across a wide range of subjects and skill levels. This platform will include user-friendly interfaces for both students and tutors to create profiles, search for matches, schedule sessions, process payments (if applicable), and provide feedback. Additionally, the project will involve implementing robust security measures to safeguard user data and privacy, as well as scalable infrastructure to accommodate a growing user base. Continuous monitoring, feedback collection, and iterative improvements will be integral to ensuring the platform's ongoing relevance and effectiveness in the education landscape.

## 1.5 Problem

The problem that the Tutor Finder project seeks to address revolves around the prevalent educational challenges faced by students today. In a world marked by diverse learning needs, varying academic levels, and geographical disparities, many students encounter difficulties in accessing high-quality educational support. Traditional tutoring methods may be inaccessible or unaffordable for some, limiting their ability to overcome academic hurdles. Moreover, the lack of a centralized and transparent platform often leads to challenges in finding qualified tutors, resulting in mismatched tutoring experiences.

## 1.6 Feasibility

The feasibility of this project is to check and verify by all resources and it can develop and used by all people. All type of feasibilities is checked and verified to develop this project.

### 1.6.1 Technical Feasibility

My project is completely feasible within the limits of current technology. Now-a-days technology is used in almost every handset. Both software and hardware are required for this website, and it integrates with a database so there won't be any technical issues regarding to feasibility.

Following are the technical elements of our project:

- HTML
- CSS
- Bootstrap
- PHP
- JavaScript
- jQuery
- MySQL

### 1.6.2 Operational Feasibility

The operational feasibility of our project depends on following elements:

- An operating system (Windows 7/8/8.1/10, Mac OS, Linux, Ubuntu, Android)
- Any hardware more than Pentium 4.
- Any Browser
- Any Mobile Device with Internet facility.

### 1.6.3 Economy Feasibility

The project is feasible by the cost, this project can be developed on the effective cost and we don't need more money on this project to develop it. This project is one-time paid cost project and we don't need to waste more money on its maintenance.

### 1.6.4 Information Feasibility

The project while based on the modern and advanced terminologies, it will provide feasibility, reliability and will be consistent.

### 1.6.5 Motivational Feasibility

Our product will be self-explanatory nature and the end user would not require intensive training session or learning material for using our website. The end user will have no confusion about whatever he/she wants to do. The graphical user interface (GUI) of product will be so attractive and user friendly that it will motivate the end user about

use of our website so that user will have a feeling of ease and contentedness while interacting with our web application.

### 1.6.6 Legal and Ethical Feasibility

This project supports legally and ethically to the entire website user. In our project there is no ethical issue as it has no copied material and any copyright violation content, and the house owner has to be registered first by giving their all details and in case of any issue that detail will be given to the concerned people and we will place an agreement for creation of bank account to get our state legally strong

## 1.7 Risk Management

Risk management is done by making a risk management strategy all the deliverables we delivered on time and done in limits constraints so that the project may not over mans and budget. Complexity is managed by providing easy and simple interface to the user

- Budget may over run.
- Validation Risks.
- Time may over run.
- Verification risks.
- Complexity management is done.

So, we handle risks through proper strategy management and scenarios. Main strong this is project work and deployment timelines. It is the best suited option for the better management of the houses. In house command and control system also help us to understand and control the working tasks and implementation done by the team. Handel the time and their values more critical option all the time. We manage it through our team work and management

# CHAPTER NO 2 BACKGROUND

In this chapter we will discuss about the sources needed for the development of the system so, first of all we analyze our system and check for the available resources and required resources

## 2.1 Area of Study

Different areas of study of computer sciences are involved in the development of this project

- Software engineering.
- Advance Programming
- SQL Database

## 2.2 Specifications

Basically, this is responsive web base application and this will run on a system. This website must be able to run on any screen and system.

## 2.3 Tools & Technologies

Coding is a process of turning program logic into specific instructions that can be executed by the computer system. In developing the project, selection of tool is the main and important factor to be considered during the development of a new system, as proposed system performance based on the tools used for its development. Behind a successful project, there lie a lot of factors and efforts. The selection of tool is really a big factor in the success of any project. The decision depends on many factors, which includes the nature of the problem, nature and amount of data, need of users or customers, and the available facilities. After the detailed studies and comparison of some common tools, analyzing the problems of old system, considering the organization needs and keeping in view all the related things this project was developed using the following tools:

- Visual Studio Code
- CSS
- Bootstrap

- HTML
- JavaScript
- jQuery
- PHP
- MySQL
- XAMPP

Following is the explanation of all these tools and how we use them in our final year project.

### 2.3.1 Visual Studio Code

Visual Studio Code is a lightweight but powerful source code editor that runs on desktop and is available for Windows, macOS, and Linux. It comes with built-in support for JavaScript, TypeScript, and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, and Go) and runtimes (such as .NET and Unity).

Aside from the whole idea of being lightweight and starting quickly, VS Code has IntelliSense code completion for variables, methods, and imported modules; graphical debugging, lining, multi-cursor editing, parameter hints, and other powerful editing features; snazzy code navigation and refactoring; and built-in source code control including Git support. Much of this was adapted from Visual Studio technology. That's why we have used Visual Studio Code for our final year project.

### 2.3.2 HTML

HTML stands for Hyper Text Markup Language. It is a standard markup language for web page creation. It allows the creation and structure of sections, paragraphs, and links using HTML elements (the building blocks of a web page) such as tags and attributes. Developers use HTML code to design how a browser displays web page elements, such as text, hyperlinks, and media files. Also they can easily navigate and insert links between related pages and websites as HTML is heavily used to embed hyperlinks.

It's also worth noting that HTML is not considered a programming language as it can't create dynamic functionality. It is now considered an official web standard. So, in HTML, we have made whole structure of our website.

### 2.3.3 CSS

CSS stands for Cascading Style Sheets, and it's used to add style to a web page by dictating how a site is displayed on a browser. CSS is unique in that it doesn't create any new elements. Like HTML or JavaScript. Instead, it's a language used to style HTML elements.

CSS is responsible for the text style, size, positioning, color, and more on a website. It's also what controls how a website's style shifts between desktop and mobile versions. Without CSS, websites would look pretty boring.

Using CSS, we have designed our website's front end intuitive and attractive. We have used external CSS and a little bit inline CSS for styling our paying guest website. We styled web pages with different CSS properties like color, border, font-family, font-size, opacity, and hover effects, etc. Also, we have made our website responsive for different devices and screens.

### 2.3.4 Bootstrap

Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive and mobile-first website. Bootstrap is a free collection of tools for creating websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Bootstrap is a framework to help you design websites faster and easier. It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels, etc. Here are some additional reasons to use Bootstrap Bootstrap's responsive CSS adjusts to phones, tablets, and desktops.

### 2.3.5 JavaScript

JavaScript is a powerful and flexible programming language. It can execute on a web browser that allows us to make interactive webpages such as popup menus, animations, form validation, etc. Where HTML and CSS are languages that give structure and style to web pages, JavaScript gives web pages interactive elements that engage a user. JavaScript can calculate, manipulate and validate data. We have used JavaScript in our project for creating

### 2.3.6 jQuery

jQuery is a lightweight, "write less, do more", JavaScript Library. It greatly simplifies JavaScript programming. jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. (jquery.com). The purpose of Query is to make it much easier to use JavaScript on your website. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

### 2.3.7 PHP

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP. PHP is a server-side scripting language that is used to develop Static or Dynamic websites or Web applications, PHP stands for Hypertext Pre-processor, that earlier Web pages. PHP is a widely-used, free, and efficient alternative to competitors such as stood for Personal Home Pages, We used PHP language in our project for back-end programming. We have done all the coding which involves inserting or retrieving data from the database using PHP. Also, we have created appropriate logics for properly managing and displaying data in our website. So, PHP was an essential tool for us to create our website in more efficient and effective way.

### 2.3.8 MySQL

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic websites.

In our project, we have used MySQL with PHP for back-end coding. We have used different commands for inserting, displaying, updating and deleting data from relational database. These commands work with PHP code for server-side programming of a website.

### 2.3.9 XAMPP

XAMPP is a cross-platform web server that is free and open-source. XAMPP is a short form for Cross-Platform, Apache, MySQL, PHP, and Perl. XAMPP is a popular cross-platform web server that allows programmers to write and test their code on a local webserver. This local host or server runs on personal computer, whether it's a desktop or a laptop. It is used to test clients or websites before publishing them to a remote web server. We have used XAMPP server for developing our project paying guest accommodation system. So, there is no need to buy web hosting for our project deployment.

## 2.4 Input Types

It is necessary to determine the various types of inputs. Inputs can be categorized as follows:

- Interactive, which are inputs entered during a dialogue.
- External inputs, which are prime inputs for the system.
- Internal inputs, which are user communications with the system.
- Operational, which are computer department's communications to the system.

## 2.5 Input Media

At this stage choice has to be made about the input media. To conclude about the input media consideration has to be given to

- Type of input
- Flexibility of format
- Speed
- Accuracy
- Verification methods
- Rejection rates
- Easy of correction
- Storage and handling requirements
- Security
- Easy to use
- Portability

Keeping in view the above description of the input types and input media, it can be said that most of the inputs are of the form of internal and interactive. As input data is to be the directly keyed in by the user, the keyboard can be considered to be most suitable input device.

## 2.6 Output Design

Computer output is the most important & direct source of information to the user. The system is accepted by the user only by the quality of its output. If the output is not of good quality, the user is likely to reject the system. Therefore, an effective output design is the major criteria for deciding the overall quality of the system.

## 2.7 Output Design Objectives

1. Designing output to serve the intended purpose.
2. Designing output to fit the user.
3. Delivering the appropriate quantity of output.
4. Making sure the output is where it is needed.
5. Providing the output on time.
6. Choosing the right output method.

## 2.8 Error Avoidance

We avoided the error in our project by following steps

- Used simple approach to code the web application.
- Used the tools which we covered throughout our degree.
- We handled time and data so that we can avoid any inconvenience.

## 2.9 Error Detection

No matter how carefully and efficiently we do our project, there will always be the possibility of an error weather it can be syntax error, logical error or runtime error. In order to fix these errors, we will run a complete diagnostic of our web application so that our web application can always work efficiently and effectively

## 2.10 Data Validation

In case of any error, we put the checks in our coding so that it can give a prompt message of what and where the error is. In this efficient way we can detect and resolve the error in our project

## 2.11 Error Message Design

The designing of error messages is an important part of the user interface design. As user is bound to commit some errors or other while designed to be helpful by providing the user with information regarding the error, he/she has committed.

# CHAPTER NO 3 SYSTEM REQUIRMENT ANALYSIS

Requirements analysis focuses on the tasks that determine the needs or conditions to meet the new or altered product or project, taking account of the possibly conflicting requirement of the various stakeholders, analyzing, documenting, validating and managing software or system requirements.

Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

## 3.1 Steps include in system requirement

### 3.1.1 Develop Requirement

The first step is to gather, analyze and develop requirements from the Concept of operations, stakeholder needs, objectives and other external requirement. Once requirements are documented, they are prioritized, de-conflicted, and validated with the stakeholders.

### 3.1.2 Write Down Requirements

The second step focuses on writing down the functional and performance requirements into the appropriate requirements documents; Initial Capabilities Document (ICD), Capability Development Document (CDD), and Capability Production Document (CPD) Requirements must be documented in order to establish requirements baseline to start building a system and manage any changes.

### 3.1.3 Check Completeness

The third step is to check that a complete set of requirements have been developed and documented that defines all system functions that are needed to satisfy the stakeholder needs with their associated performance, environmental, and other non-functional requirements. Requirement Tracing is a big tool in this step.

### 3.1.4 Analyze, Refine, and Decompose Requirements

Requirements Analysis is the first major step in the Systems Engineering Process. This step examines each requirement to see if it meets the characteristics of a good requirement. Each requirement is then decomposed into a more refined set of requirements that are allocated to sub-systems and documented in the Weapons System Specification (WSS). Newly derived requirements are expected to emerge from this process, which continues until all requirements are defined and analyzed.

### 3.1.5 Validate Requirements

In this step each requirement must be verified and validated to ensure that these are the correct requirements. This ensures that the requirements meet the overall objective of the system and all stakeholder needs.

### 3.1.6 Manage Requirements

In this step the requirements have been accepted and a baseline is established by the stakeholders. Any changes to the requirements are controlled using a Configuration Management process.

## 3.2 Functional Requirement

A Functional Requirement (FR) is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputs to the software system, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform.

Functional requirement is the user expects from the software for example if the application is a blank application that application should be able to create a new account, update the account, delete an account, etc. functional requirement is detailed and are specified in the system design. Functional requirement: specify the functionality of the system (E.g., field in a form). A functional requirement describes what a software should do. Functional requirements explain what has to be done by identifying the necessary task, action or activity that must be accomplished.

Our project has two modules.

1. Admin Module
2. Tutor Module

### 3.2.1 Admin Module

1. Manage dashboard
2. See requests
3. Accept/Reject Tutor Request
4. Add and remove courses
5. Read/View inquires
6. Manage Tutor Profile

### 3.2.2 Tutor Module

1. Can create account (sign up, log in)
2. Manage profile
3. Add Subjects
4. Check Inquiries
5. Check the account status
6. Change password or email
7. Logout

## 3.3 Non-Functional requirement

It specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. The non-functional requirements elaborate a performance characteristic of a system. Non-functional requirements specify the quality of a system and is mostly related the satisfiability of the user. Our project has following non-functional requirements.

- **Efficiency**

  When this paying accommodation portal system will be implemented users and house owners can easily access online accommodation system and searching for accommodation will be very faster.

- **Reliability**

  The system should accurately perform accommodation registration, user validation, report generation and search.

- **Usability**

  The system is designed for a user-friendly environment so that student and house owners can perform the various tasks easily and in an effective way.

Other non-functional requirements of our project fall into following areas:

- Accessibility
- Compliance
- Extensibility
- Fault tolerance
- Maintainability
- Privacy
- Scalability
- Security
- User Friendly
- Flexibility
- User Safety

## 3.4 Software Quality Attributes

The quality of the system is maintained in such a way so that user as friendly environment.

The software quality attributes are assumed as under.

- Accurate and reliable
- Secure
- Fast speed
- Compatibility

# CHAPTER NO 4 SYSTEM DESIGN

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. System design is also defined as the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

We have included the following artifacts in our system design approach:

1. SDLC
2. General Model
3. Data Flow Diagram
4. Use Case Diagram
5. ER Diagram
6. Tables
7. Class Diagram

## 4.1 SDLC

The Software Development Life Cycle (SDLC) refers to a methodology with clearly defined processes for creating high-quality software. in detail, the SDLC methodology focuses on the following phases of software development:

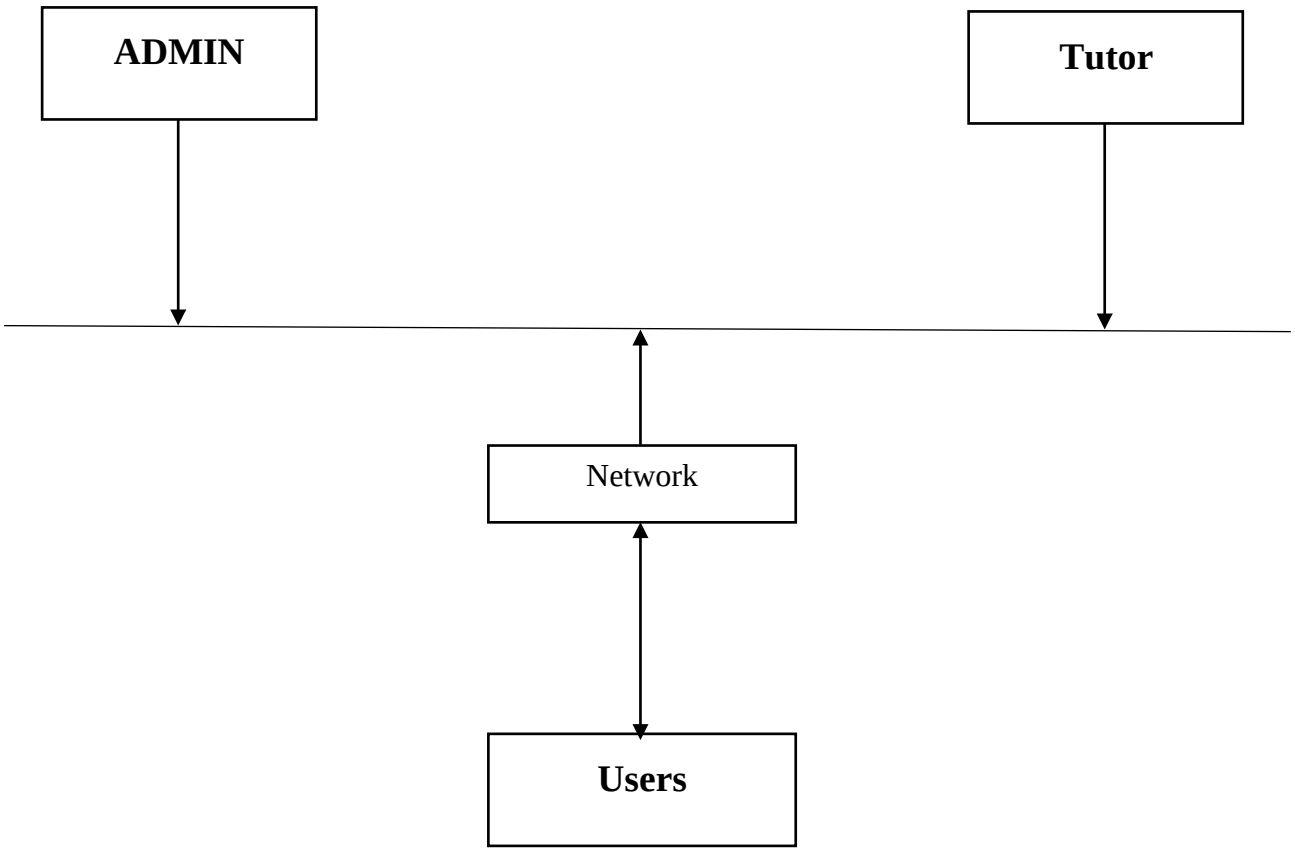
- Requirement analysis
- Planning
- Software design such as architectural design
- Software development
- Testing
- Deployment

SDLC or the Software Development Life Cycle is a process that produces software with the highest quality and lowest cost in the shortest time possible. SDLC provides a well-structured flow of phases that help an organization to quickly produce high-quality software which is well-tested and ready for production use

Real project rarely follows the sequential flow. In general, these phase overlap and feed information to each other. Hence there should be an element of iteration feedback. A mistake caught at any stage should be referred back to the source and all subsequent stages need to be visited and corresponding documents should be updated accordingly. Waterfall model is a documentation-driven model. It therefore generates complete and comprehensive documentation and hence makes the maintenance task much easier. Our project followed these methodologies and phases of development.

## 4.2 General Model

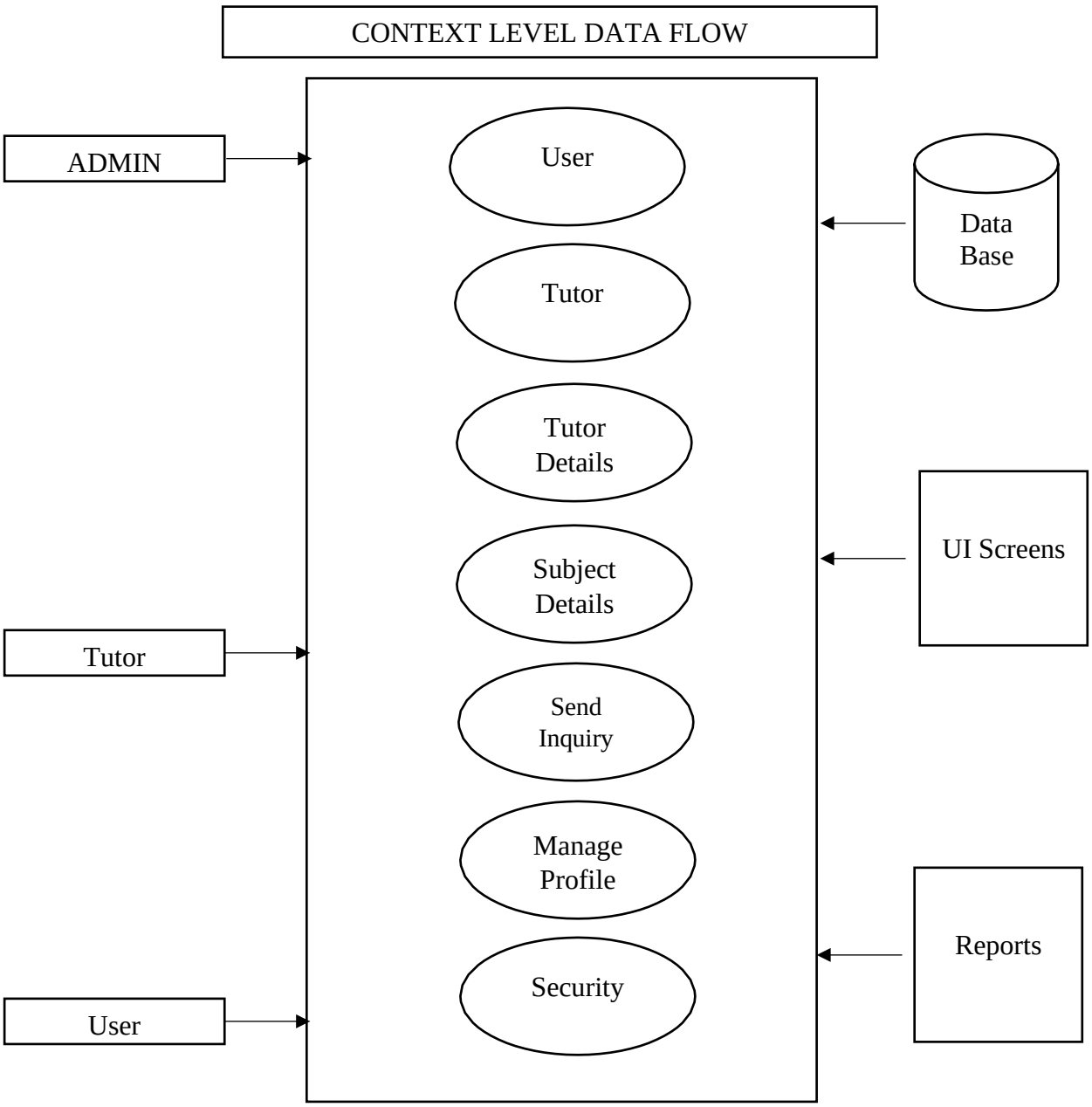
General model of online accommodation is shown in fig in which administrator and staff at remote Server can send request to client system to capture their desktop.



## 4.3 Data Flow Diagram (DFD)

The system is divided into various modules, each module is further divided into sub-module. The connection of the main modules is shown with the help of DFDs. DFDs are made of the following representation.

### 4.3.1 Data Flow Diagram for Context Level



### 4.3.2 Data Flow Diagram for User Level-1

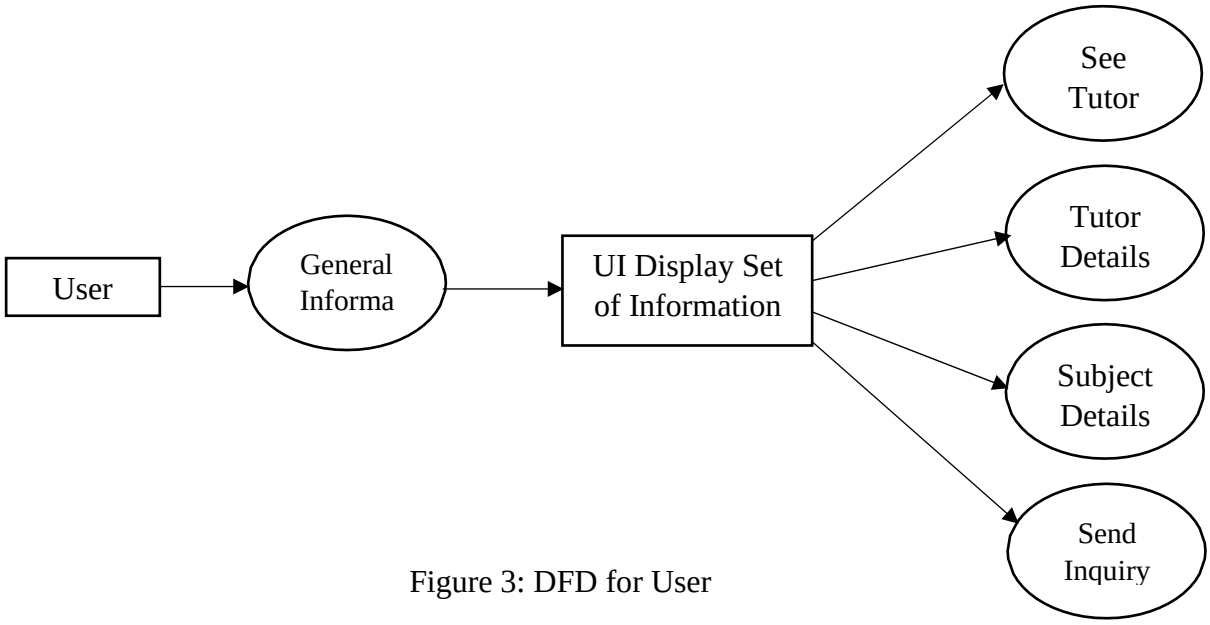
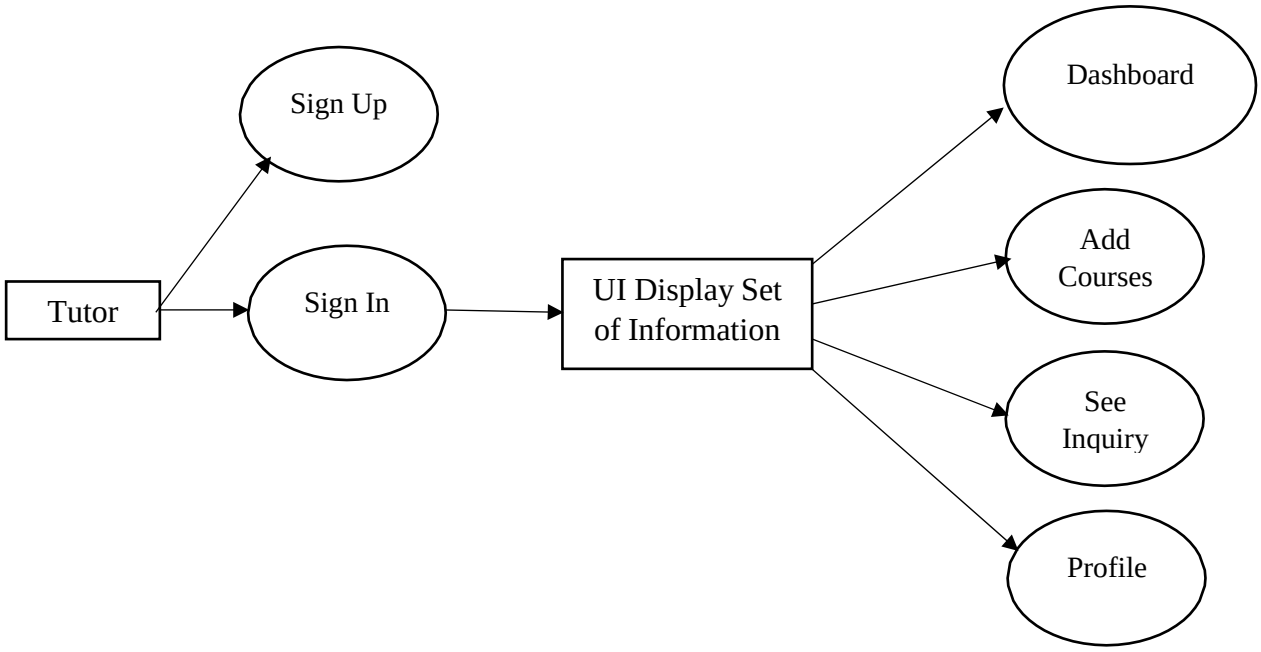
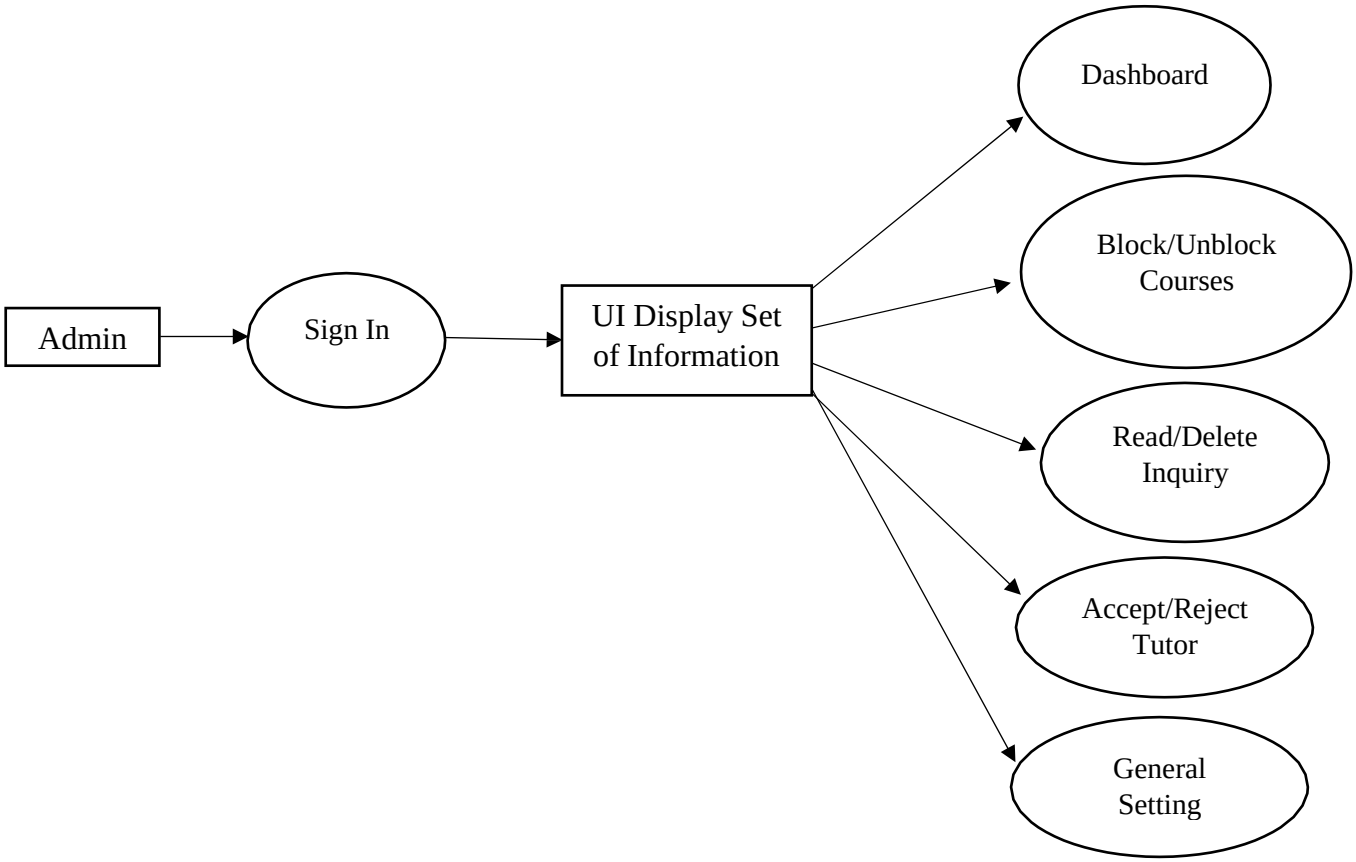



Figure 3: DFD for User

### 4.3.3 Data Flow Diagram for Tutor Level-1



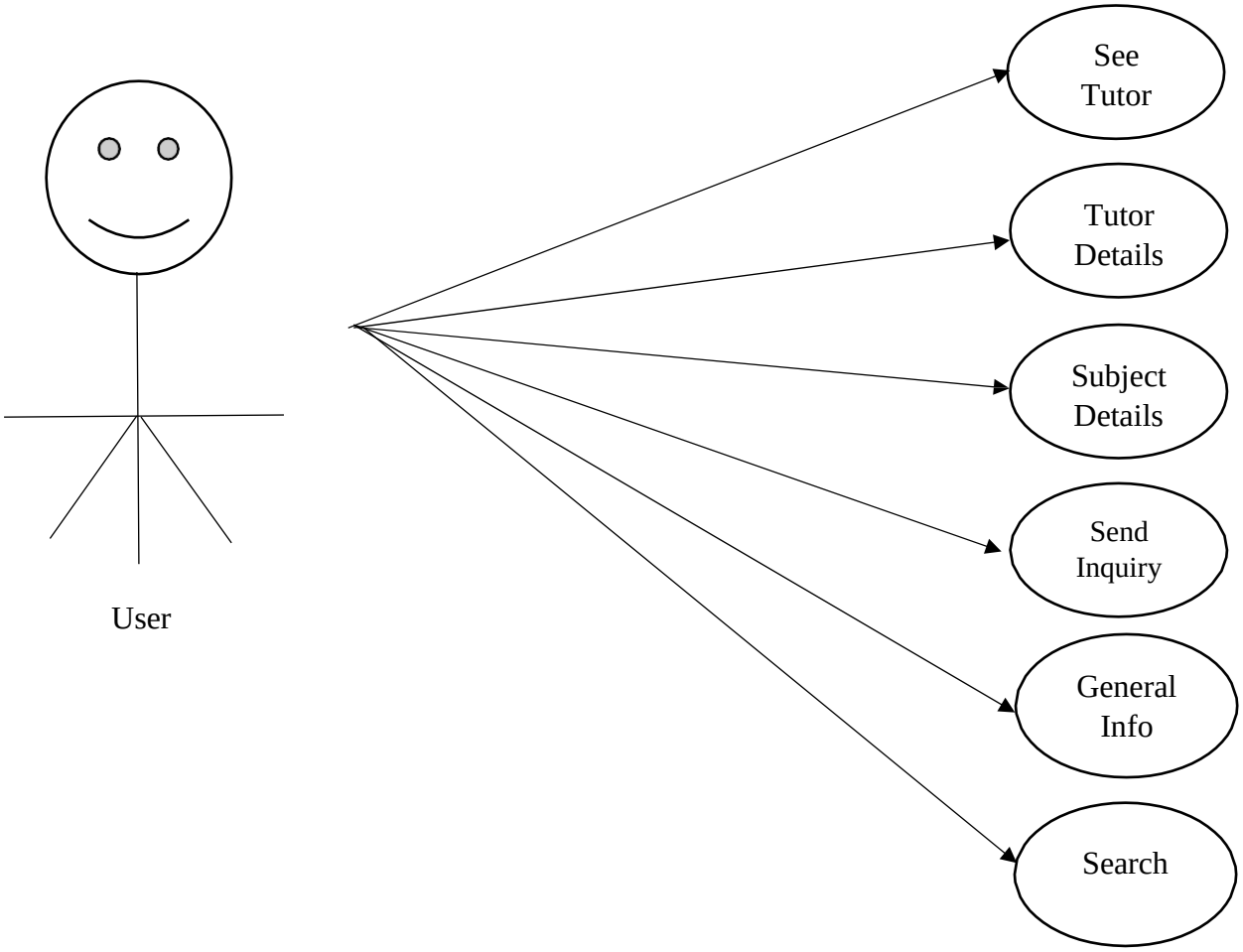
### 4.3.4 Data Flow Diagram for Admin Level-1



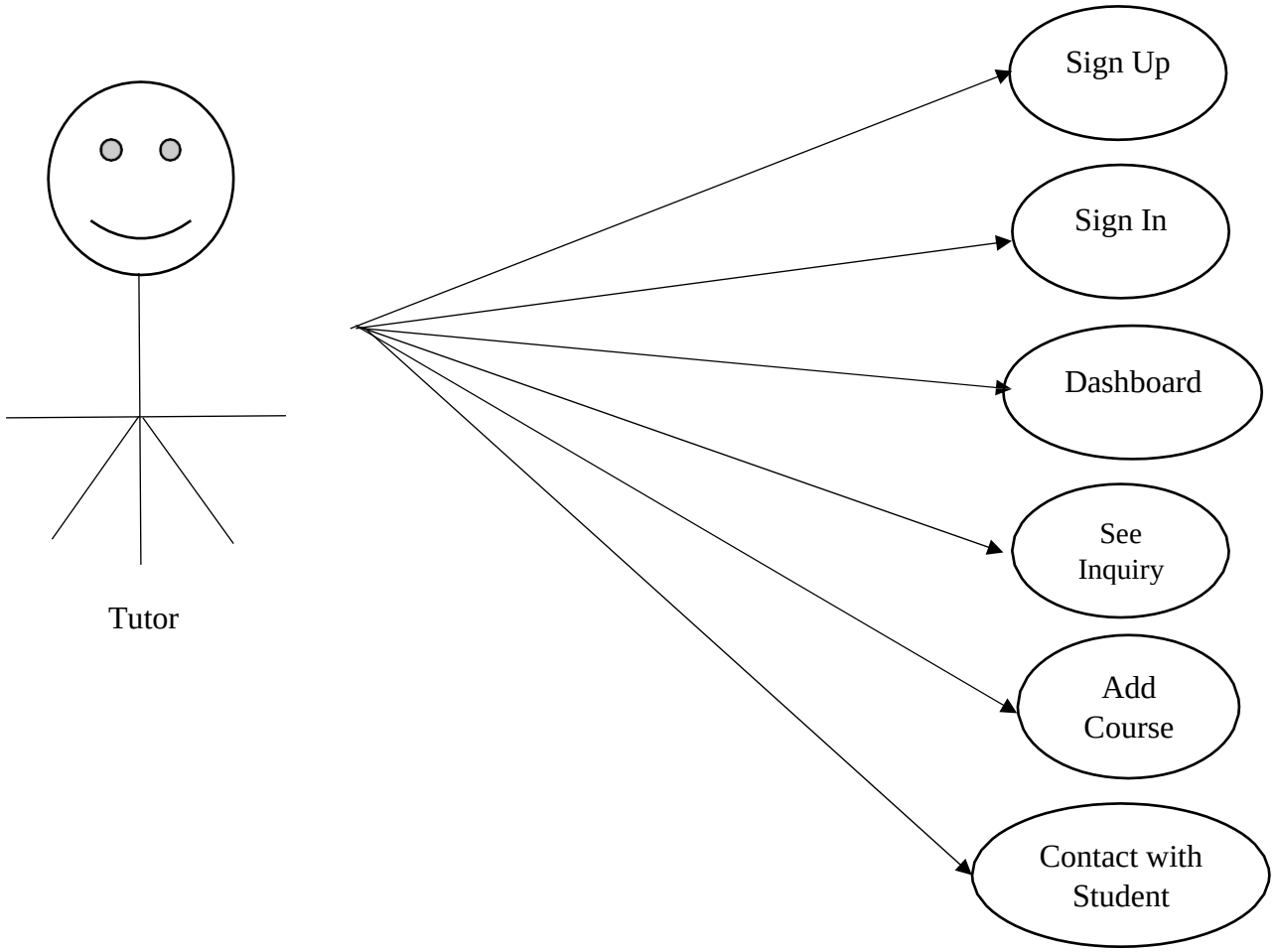
## 4.4 Use Case Diagrams

Use case diagram consists of actors, use cases and their relationships. These diagrams are especially important in organizing and modelling the behavior of a system.

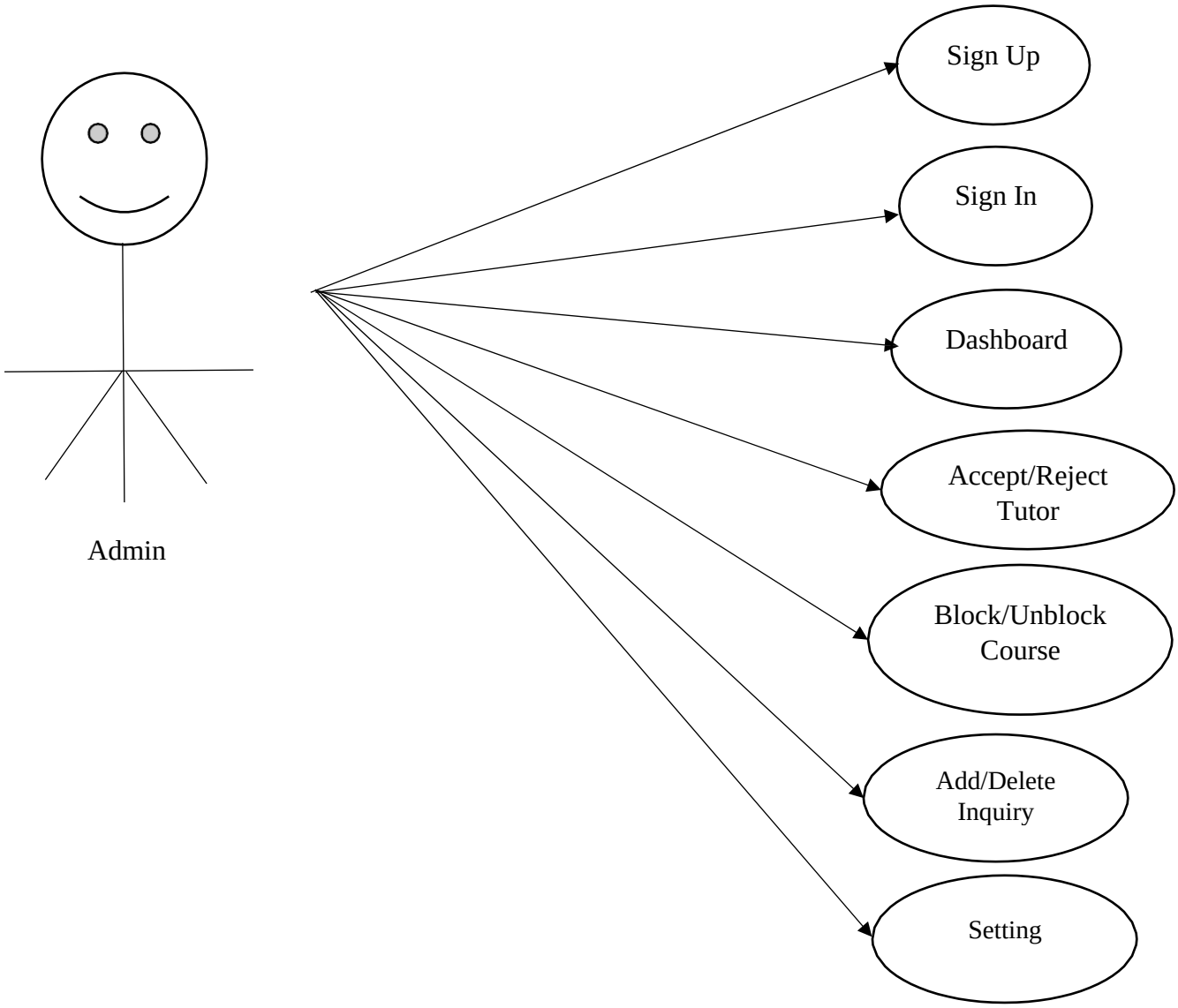
### 4.4.1 User Use Case Diagram



### 4.4.2 Tutor Use Case Diagram



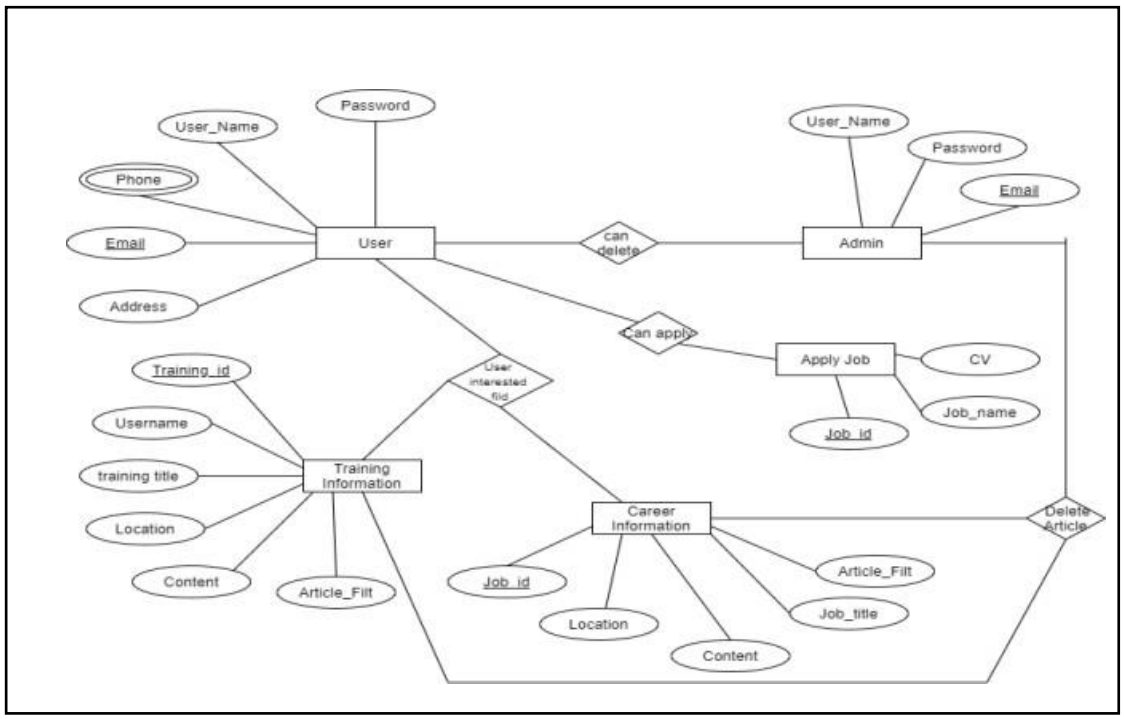
### 4.4.2 Admin Use Case Diagram



## 4.5 ER Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD that displays the relationship of entity sets stored in a database. In other words, ER diagram help to explain the logic structure of database. ER Diagrams are created based on three basic concepts: entities, attributes and relationships. ER Diagram contain different symbols that use rectangular to represent entities, ovals to define attributes and diamond shapes to represent relationships.

Following is the ER Diagram of our project Tutor Finder:



## 4.6 Tables

Tables are database objects that contain all the data in a database. In tables, data is logically organized in a row-and-column format similar to a spreadsheet. Each row represents a unique record, and each column represents a field in the record.

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MySQL. Access database has been chosen for developing the relevant databases. Pet's Mania Project contains 8 MySQL tables.

### 4.6.1 Table for Admin

**tbladmin:** This table store the admin login details

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(50) | | | No | *None* | | AUTO_INCREMENT | Change Drop More |
| 2 | firstname | varchar(250) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| 3 | middlename | text | utf8mb4_general_ci | | Yes | *NULL* | | | Change Drop More |
| 4 | lastname | varchar(250) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| 5 | username | text | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| 6 | password | text | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| 7 | avatar | text | utf8mb4_general_ci | | Yes | *NULL* | | | Change Drop More |
| 8 | last_login | datetime | | | Yes | *NULL* | | | Change Drop More |
| 9 | type | tinyint(1) | | | No | 0 | | | Change Drop More |
| 10 | date_added | datetime | | | No | current_timestamp() | | | Change Drop More |
| 11 | date_updated | datetime | | | Yes | current_timestamp() | | ON UPDATE CURRENT_TIMESTAMP() | Change Drop More |

### 4.6.2 Table for Tutor

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(30) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | firstname | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 3 | middlename | text | utf8mb4_general_ci | | Yes | NULL | | | Change Drop More |
| 4 | lastname | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 5 | email | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 6 | password | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 7 | avatar | text | utf8mb4_general_ci | | Yes | NULL | | | Change Drop More |
| 8 | status | tinyint(1) | | | No | 0 | 0 = Not Validated, 1 = Validated, 2 = Inactive, 3 = Blocked | | Change Drop More |
| 9 | delete_flag | tinyint(1) | | | No | 0 | | | Change Drop More |
| 10 | date_created | datetime | | | No | current_timestamp() | | | Change Drop More |
| 11 | date_updated | datetime | | | No | current_timestamp() | | ON UPDATE CURRENT_TIMESTAMP() | Change Drop More |

### 4.6.3 Table for Course

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(30) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | tutor_id | int(30) | | | No | None | | | Change Drop More |
| 3 | name | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 4 | description | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 5 | experience | varchar(150) | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 6 | status | tinyint(1) | | | No | 1 | | | Change Drop More |
| 7 | logo_path | text | utf8mb4_general_ci | | Yes | NULL | | | Change Drop More |
| 8 | delete_flag | tinyint(1) | | | No | 0 | | | Change Drop More |
| 9 | date_created | datetime | | | No | current_timestamp() | | | Change Drop More |
| 10 | date_updated | datetime | | | No | current_timestamp() | | ON UPDATE CURRENT_TIMESTAMP() | Change Drop More |

### 4.6.4 Table for Inquiry

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(30) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | tutor_id | int(30) | | | No | None | | | Change Drop More |
| 3 | course_id | int(30) | | | No | None | | | Change Drop More |
| 4 | fullname | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 5 | email | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 6 | contact | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 7 | message | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 8 | status | tinyint(1) | | | No | 0 | | | Change Drop More |
| 9 | date_created | datetime | | | No | current_timestamp() | | | Change Drop More |
| 10 | date_updated | datetime | | | No | current_timestamp() | | ON UPDATE CURRENT_TIMESTAMP() | Change Drop More |

### 4.6.5 Table for System Information

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(30) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | meta_field | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 3 | meta_value | text | utf8mb4_general_ci | | No | None | | | Change Drop More |

# CHAPTER NO 5 TESTING AND IMPLEMENTATION

## 5.1 Introduction

System Analysis and Design process including Requirement Analysis, Business Solution Options, Feasibility Study, and Architectural Design. Generally, Software bugs will almost always exist in any software module. But it is not because of the carelessness or irresponsibility of programmer but because of the complexity. Humans have only limited ability to manage complexity.

## 5.2 Testing

Testing is the process of executing a program or system with the intent of finding errors. The scope of software testing often includes examination of code as well as execution of that code in various environments and conditions.

The aim of the system testing process was to determine all defects in our project. The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not. Testing stages of the project can be explained as below and system was tested for all these stages.

### 5.2.1 Unit Testing

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module, we need to provide a complete environment that is besides the module we would require.

The procedures belonging to other modules that the module under test calls non local data structures that module accesses. A procedure to call the functions of the module under test with appropriate parameters. Unit testing was done on each and every module.

**Test for the Admin Module**

Testing admin login form-This form is used for log in of administrator of the system. In this we enter the username and password if both are correct administration page will

open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.

**House Module**

In this section the user can get details from main database and the whole information depending upon the user specification is generated as a table.

### 5.2.2 Integration Testing

After each unit is thoroughly tested, it is integrated with other units to create modules or components that are designed to perform specific tasks or activities. These are then tested as group through integration testing to ensure whole segments of an application behave as expected. These tests are often framed by user scenarios, such as logging into an application or opening files. Integrated tests can be conducted by either developers or independent testers and are usually comprised of a combination of automated functional and manual tests. In this type of testing, we test various integration of our project module by providing the input. The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

### 5.2.3 System Testing

System testing is a black box testing method used to evaluate the completed and integrated system, as a whole, to ensure it meets specified requirements. The functionality of the software is tested from end-to-end and is typically conducted by a separate testing team than the development team before the product is pushed into production.

During system testing, the system is used experimentally to ensure that the software doesn't fail, i.e. it will run according to its specifications and in the way users expect, also special test data input for processing, and the results examined. A limited number of users may be allowed to use the system can see whether they try to use it in unforeseen ways,

### 5.2.4 Acceptance Testing

Acceptance testing is formal testing based on user requirements and function processing. It determines whether the software is conforming specified requirements and user requirements or not. It is conducted as a kind of Black Box testing where the number of required users involved testing the acceptance level of the system. It is the fourth and last level of software testing. However, the software has passed through three testing levels (Unit Testing, Integration Testing. System Testing) But still there are some minor errors which can be identified when the system is used by the end user in the actual scenario.

## 5.3 Testing Methods and Comparison

### 5.3.1 Black Box Texting

Black Box Testing is testing without the knowledge of the internal workings of the item being tested. When black box testing is applied to software engineering, the tester selects valid and invalid input and what the expected outputs should be, but not how the p program actually arrives at those outputs. Black box testing methods include equivalence partitioning. Boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing. This method of test design is applicable to all levels of software testing unit, integration, functional testing, system and acceptance

### 5.3.2 White Box Texting

White box testing (glass box testing) strategy deals with the internal data structures and algorithms. The tests written based on the white box testing strategy incorporate coverage of the code written, branches, paths, statements and internal logic of the code, etc. These testers require programming skills to identify all paths through the software. Types of white box testing includes code coverage (creating tests to satisfy some criteria of code coverage), mutation testing methods, fault injection methods, static testing.

## 5.4 Testing Strategy and Test Plan

Black box testing and white box testing were used as the main testing techniques to test the entire system for all the stages. Although both of these testing techniques have advantages and disadvantages as discussed above, but when combined, they help to ensure thorough testing of the product.

All the functions, procedures are tested as single individual units first and after integrating individual units into one-unit, integrated units were tested as a whole. When the function/procedure is tested for the first time, black box testing methodology was used. It was tested for expected results on different user inputs. If the function did not deliver the expected outcome white box testing methodology was used. So it was tested by going through each and every relevant code. After testing for functionality of the functions/Procedures, these functions/procedures were combined and tested using black box testing methodology. If the results show any discrepancy than expected output, then again, they were tested for expected results using white box testing methodology. Once the project was completed then it was tested using test data by using black box and white box approach. All the interfaces were checked using black box approach. Since validation rules have been applied at all the possible areas of user interfaces, testing of some areas were not required.

Further it was checked whether the software can handle large quantities of data by loading dummy data into the system. In case of data retrieve from database it is assured that the correct data is retrieved. When data is stored in the database, it is also tested. When deleting and updating the records in tables, it was checked whether the correct row is updated.

## 5.5 Test Cases

A test case is the set of steps that need to be done in order to test a specific function of the website. They are developed for various scenarios so that testers can determine whether the website is working the way it should and producing the expected results.

### 5.5.1 Data and Database Integrity Testing

*Table 1: Data integrity Testing*

| | | |
|---|---|---|
| **Test Objective** | Our main objective of this is to ensure that the tables we created rightly Objective accessed from the user interfaces and rightly updated, modified and deleted data from the tables | |
| **Test Cases** | **Test Case Description** | **Result** |
| | Provide login information | Data entered Successfully |
| | Select a specific task and complete it | Task has been completed |
| | Add all queries related to booking, uploading tutor details, cancellation and confirmation of booking | All data has successfully entered |
| **Completion criteria** | All tables that created are successfully accessed from the user interfaces and successfully updated, modified and deleted. | |

### 5.5.2 Function Testing

*Table 2: Function Testing*

| | | |
|---|---|---|
| **Test Objective** | Our main objective of this is to test function that are on a user/owner/admin interface that fulfilling their functionality data entry, and retrieval. | |
| **Test Cases** | **Test Case Description** | **Result** |
| | Check all buttons that helps user to navigate from one page to another | All the buttons successfully performed their functions. |
| | Check login | Valid person successfully logged in. |
| | All data entry operations can be checked by entering data. | All data entry operations successfully performed their functionalities. |
| | Check any other function. | All other functions performed their functionalities. |
| **Completion** | All functions successfully performed their functionalities. | |

### 5.5.3 Tutor Interface Testing

*Table 3: User interface Testing*

<table>
<tr><td>Test Objective</td><td colspan="2">Our main objective of this to do check navigation including window to window, field to field, and use of access methods e.g., focus on other characteristics like size, position, state, color and any other characteristics</td></tr>
<tr><td rowspan="2">Test Cases</td><td>Test Case Description</td><td>Result</td></tr>
<tr><td>Every service provider like user and admin can be checked one by one.</td><td>All characteristics that are mentioned above are fulfill</td></tr>
<tr><td>Completion Criteria</td><td colspan="2">Tab key, ese key work properly, Mouse movement is ok.<br>Colors are professional and text color is ok.<br>Text is easily readable and navigation among pages is ok.</td></tr>
<tr><td>Special Consideration</td><td colspan="2">Attractive interface</td></tr>
</table>

### 5.5.4 Performance Testing

*Table 4: Performance Testing*

<table>
<tr><td>Test Objective</td><td colspan="2">Main goal of this is that when admin add, modify and delete data and place a new data the response showed in a proper time or not.<br><br>What happened when a lot of data entered at the same time to server?</td></tr>
<tr><td rowspan="4">Test Cases</td><td>Testing case Description</td><td>Result</td></tr>
<tr><td>When a lot number of request come from website</td><td>This website has its own server so all these requests should be fulfilled in proper time.</td></tr>
<tr><td>When admin enter, delete or modify any entity.</td><td>Database server respond quickly</td></tr>
<tr><td>Criteria</td><td colspan="2">There is no restriction for data. Data can be easily modified.</td></tr>
<tr><td>Special</td><td colspan="2">The database server should have capability to respond at proper time duration.</td></tr>
</table>

### 5.5.5 Security and Access Control Testing

*Table 3: Security and Access Control Testing*

| | | |
|---|---|---|
| **Test Objective** | Our main objective of this is to check's system and access control testing | |
| **Test Cases** | **Test Case Description** | **Result** |
| | Errors during viewing data by admin | There is no error, so, no need to worry. It's OK |
| **Completion Criteria** | Nothing special consideration for this except administrator. | |
| **Special Consideration** | Nothing | |

## 5.6 Implementation

System implementation projects are long difficult journeys by which organization move from an old set of technology/methods/procedures to a new one. A software implementation method is a systematic structured approach to effectively integrate software-based service or component into the workflow of an organizational structure or an individual end-user. The complexity of implementing product software differs on several issues. Examples are the number of end users that will use the product software, the effects that the implementation has on changes of tasks and responsibilities for the end user, the culture and the integrity of the organization where the software is going to be used and the budget available. It is vital to select the right strategy for implementing the application to assure successful results.

### 5.6.1 Implementation Strategy

Since the software application consists of three modules as per in the high-level architectural design, the implementation was done using iterative, incremental approach. Phase wise implementation process enables to execute by incrementally aligning the product with the end-user.

### 5.6.2 Direct Implementation

With this method of implementation, the users stop using the manual system and start using the computer system from a given date. The advantage of this method is that it is less costly in effort and time than any other method of implementation. The disadvantage of this method is that if problems occur the users do not have any alternative apart from returning to a manual system which may prove difficult if it has been discontinued.

### 5.6.3 Parallel Running

With parallel running, the new system is introduced alongside the existing system. With parallel running both systems (manual and computer, or old computer and new computer system) will be in operation at the same time.

# CHAPTER NO 6 USER MANUAL

## 6.1 Website Front View

Following are some pages of the front view of our website. When a user come to our website, he can visit these pages.

### 6.1.1 Nav Bar


- **TutorFinder**

  This is website name, to click on it, return to main home page.
- **HOME**

  Contains Home page.
- **TUTOR**

  Contains Tutor information.
- **ABOUT**

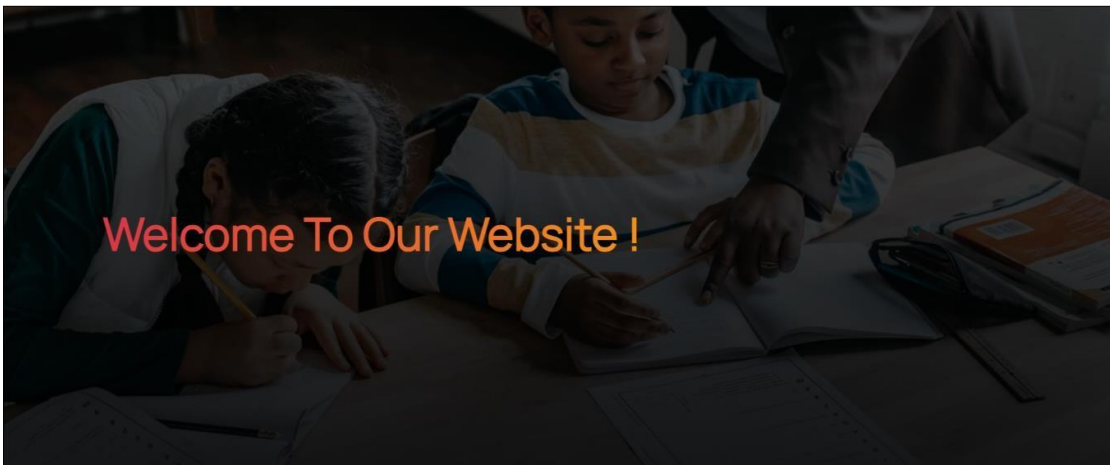
  Contains About Us information.
- **LOGIN**

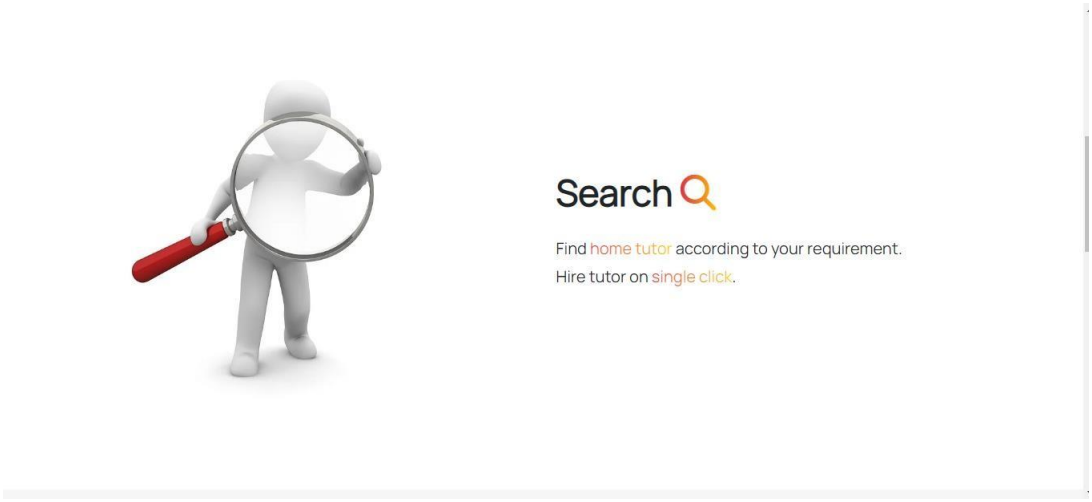
  For purpose if Login into your account.
- **Register as Tutor**

  For purpose of Sign Up into website.
- **Need a Help**

  For purpose of any help about this website.

### 6.1.2 Home Page



## Select

View tutors profile and hire a tutor who suits you best according to your need.

## Study

TutorFinder handles the behind the scenes stuff so you can focus on your study.

### 6.1.3 Footer


- **Home**

  Contains Home page.
- **About Us**

  Contains About Us page.
- **Contact Us**

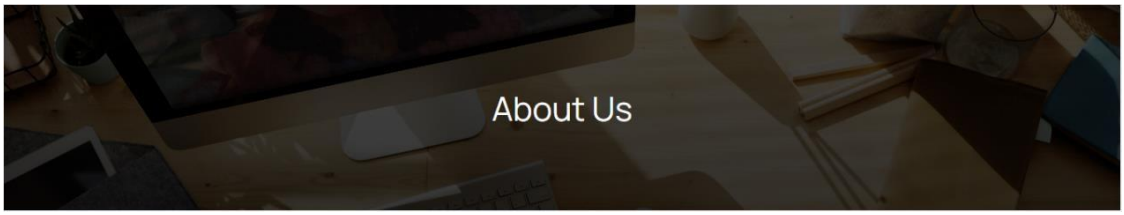
  Contain contact information.
- **FAQs**

  Contains frequently asked questions about website.
- **T&C for Tutors**

  Contain information about terms and conditions about tutors.
- **T&C for Students**

  Contain information about terms and conditions about students.

### 6.1.4 About Us Page




## INTRODUCTION

TutorFinder assume that if someone invest in studies he or she will get better return of that investment. We help students to search the best tutor in their area.We want their bright future so that they can achieve their goals.Their dreams are our dreams.If student want to achieve higher grades in studies then we are here to help them.We have set this platform for parents who take care of their child study.We believe all the students and tutors will get benifits from our site.

## OUR PURPOSE

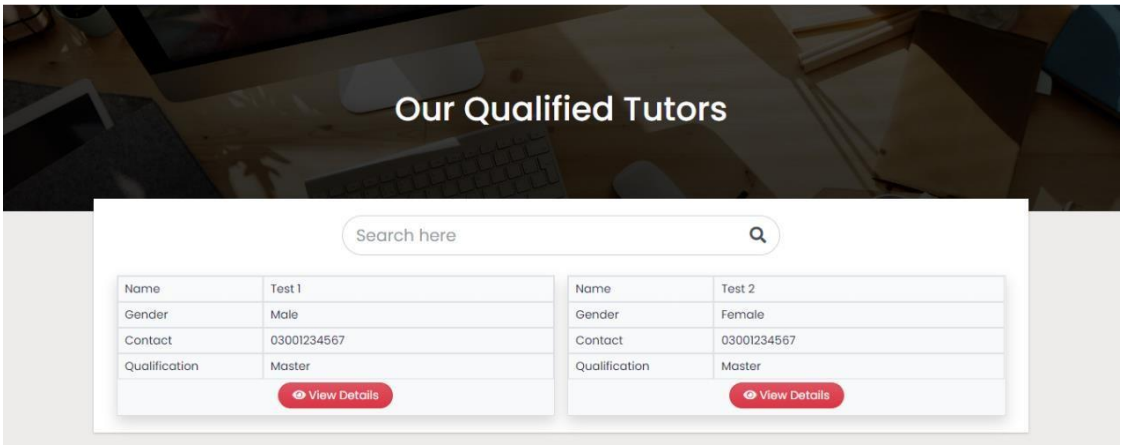
Our site create oppourtunities for tutors who can serve their skills at TutorFinder. We had started this site for overcome the difficulties such like If some one had hired home tutor and they had no trust on tutor because the academy in which that tutor is registered doesn't exist.One more claws is that In case of local academies , you don't have any option for selecting tutor according to your requirements so that's why we have set up this platform to give ease to students and parents so that they can find a home tutor which suits them best.

## BENIFITS FOR TUTORS

- Local academies charge upto 70% commission from first fee but TutorFinder will never take any commision from tutors.

## 6.2 User Module

### 6.2.1 View Tutors

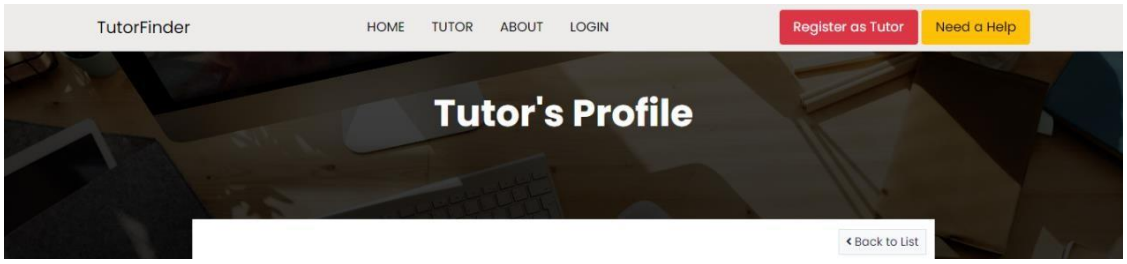


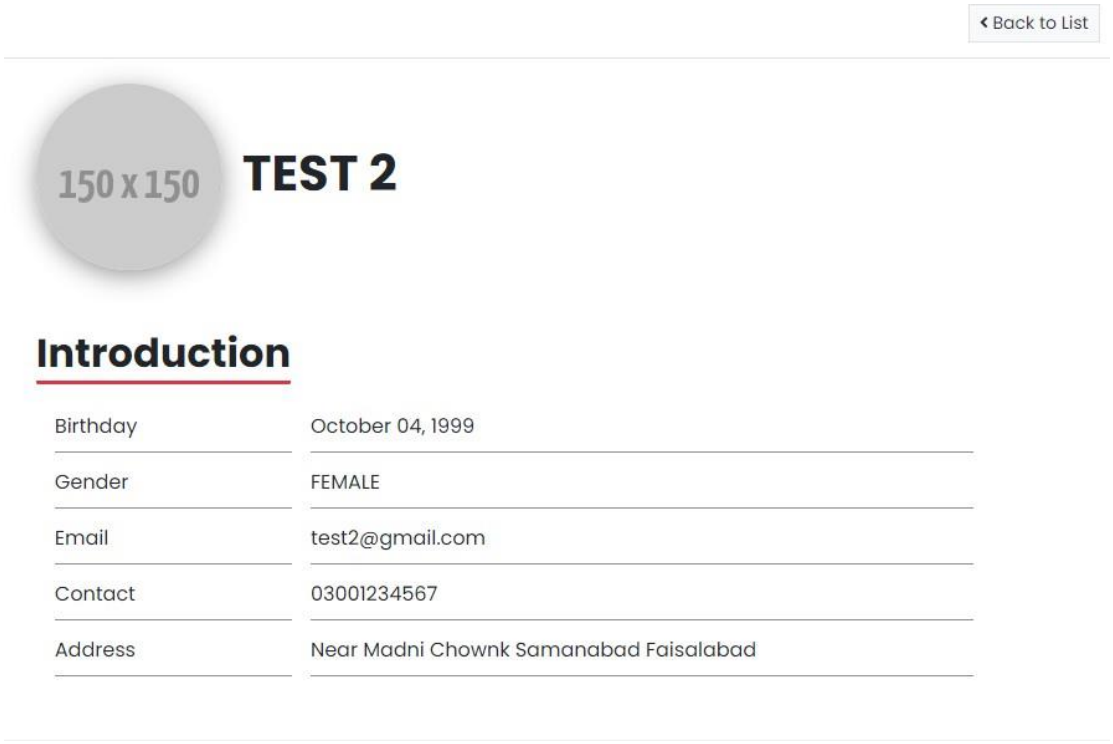

This page contains the information about our qualified tutors.

Every **Tutor Card** has some information:

- Tutor Name
- Tutor Gender
- Contact
- Qualification
- View Details

### 6.2.2 View Tutors Details



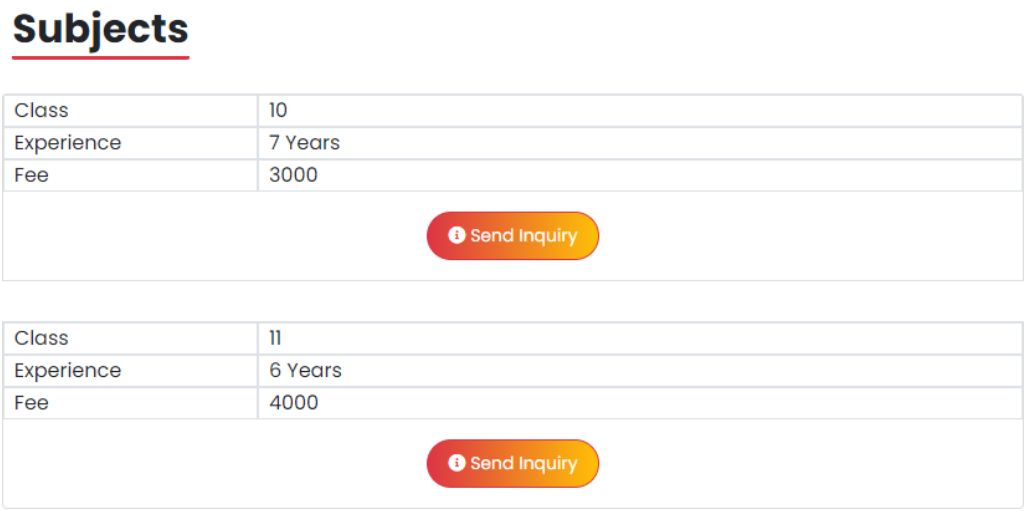




In this section user can see the information about tutor.

- Profile Picture
- Profile Name
- Birthday
- Gender
- Email
- Contact No
- Address

### 6.2.3 Send Inqurey

This is second part of Tutor Card, where a user/student can send request against their subject/class.

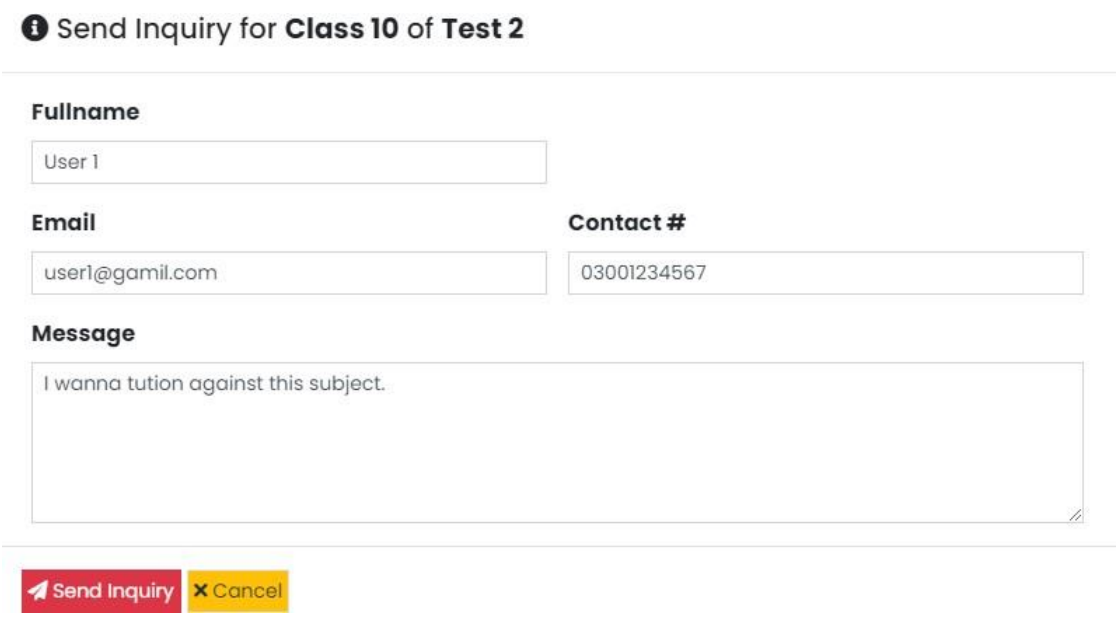



Here, some piece of information:

- Subject
- Experience
- Fee
- Send inquiry

  Click on button to send request againt their demanded subject to teacher.

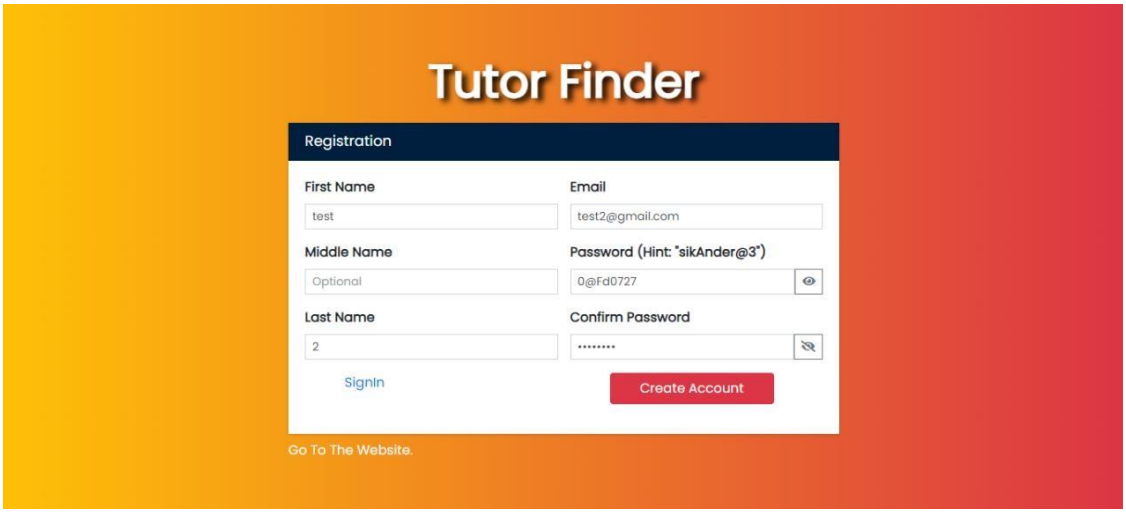
### 6.2.4 Send Request/Inquiry




- Full Name
- Email
- Contact No
- Message
- Send Inquiry
- Cancel

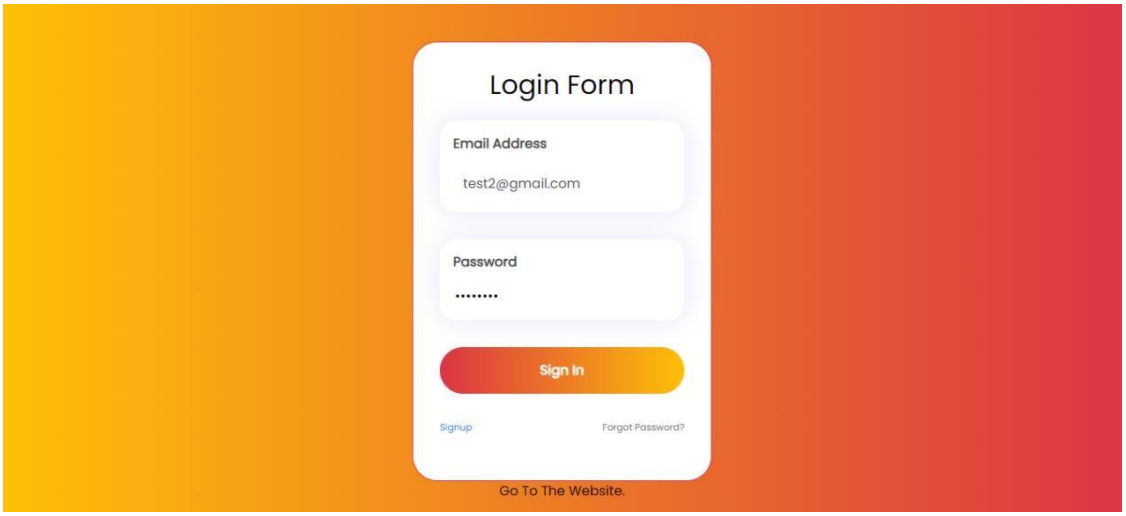
## 6.3 Tutor Module

### 6.3.1 Registration Form




- First Name (Compulsory)
- Middle Name
- Last Name
- Email
- Password
- Confirm Password
- Create Account
- Already Have an Account? SIGNUP
- Go To Website

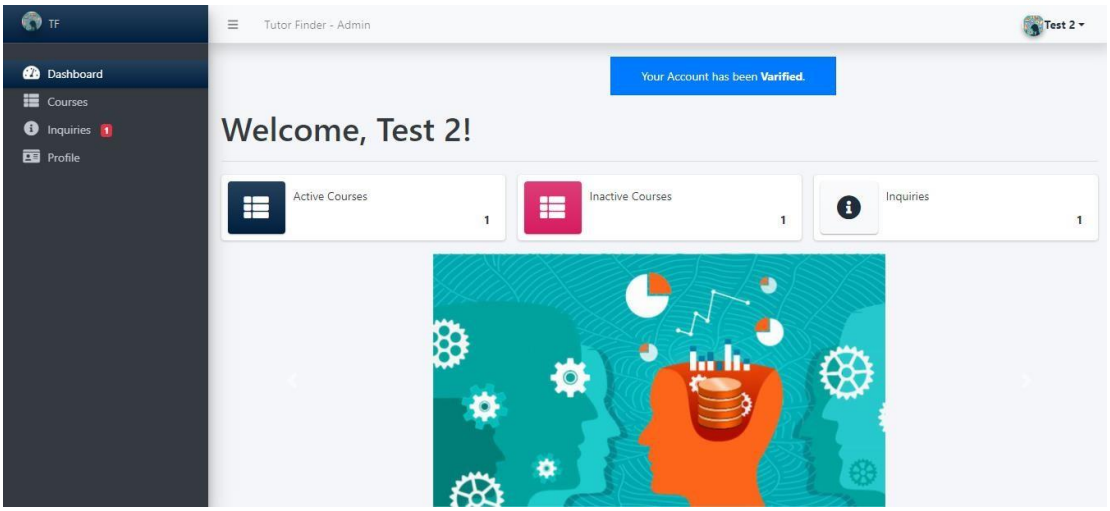
### 6.3.2 Login/Sign In




This page for access in dashboard.

- Email
- Password
- Sign In
- Have no Account? Sign Up

### 6.3.3 Dashboard




This page contains the all settings for tutors.

- **Account Status**

  This will show the pending, active and block status.

- **Dashboard**

  View all information.

- **Courses**

  Add/Remove courses.

- **Inquiries**

  Read/Delete inquiries.

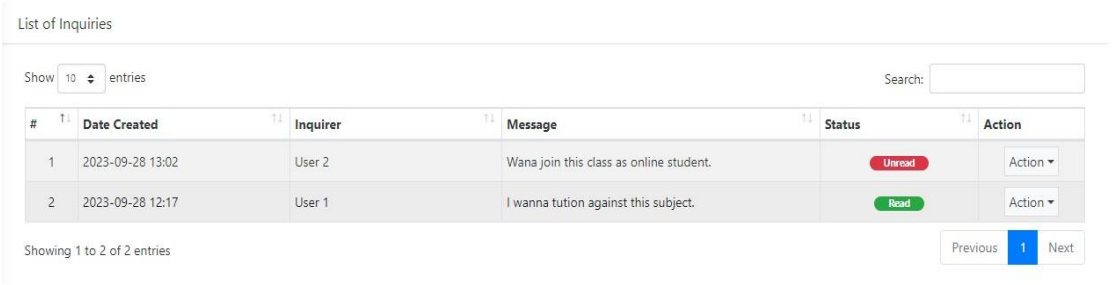
- **Profile**

  Modify information.

### 6.3.4 List of Courses

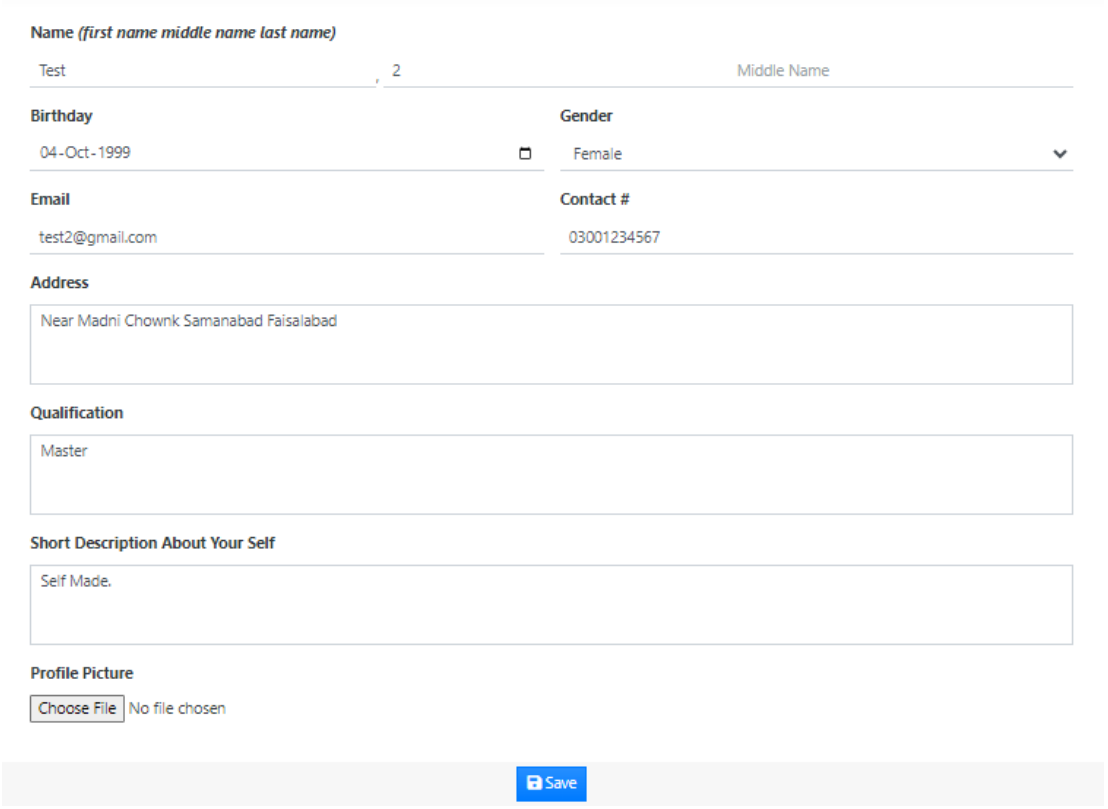

This page contains information about list of course(s). Here, tutor add their courses/subject and shown in their profile. When status will be active, it means that this course/subject will be shown in their profile and vice versa.

### 6.3.5 List of Inquiries




This page contains information about list of inquiry(s). Here, tutor see inquiries against their courses/subject and shown in their profile. When status will be read, it means that inquiry against this course/subject will be read and vice versa.

### 6.3.6 Profile

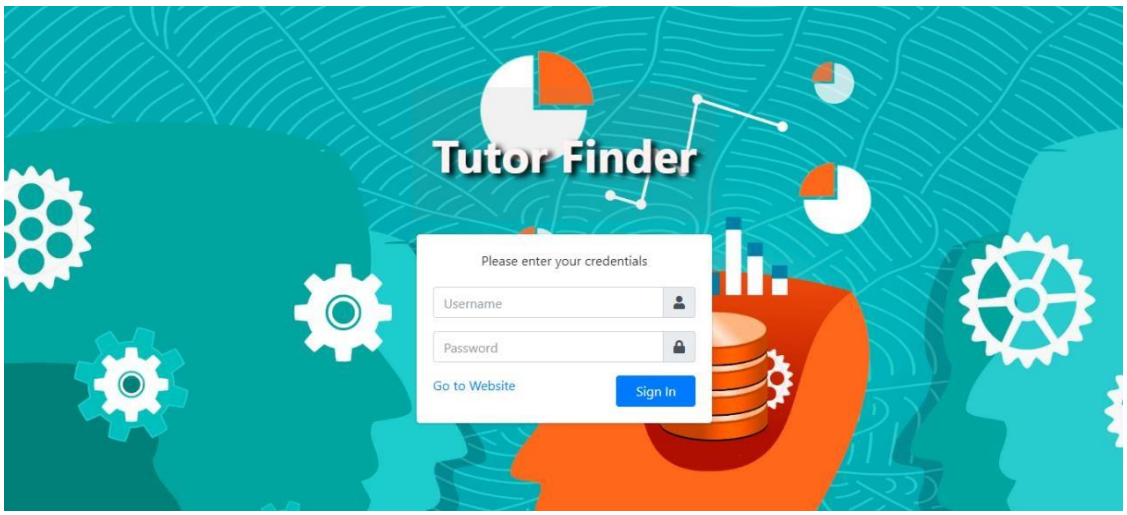



Here, tutor can modify their profile.

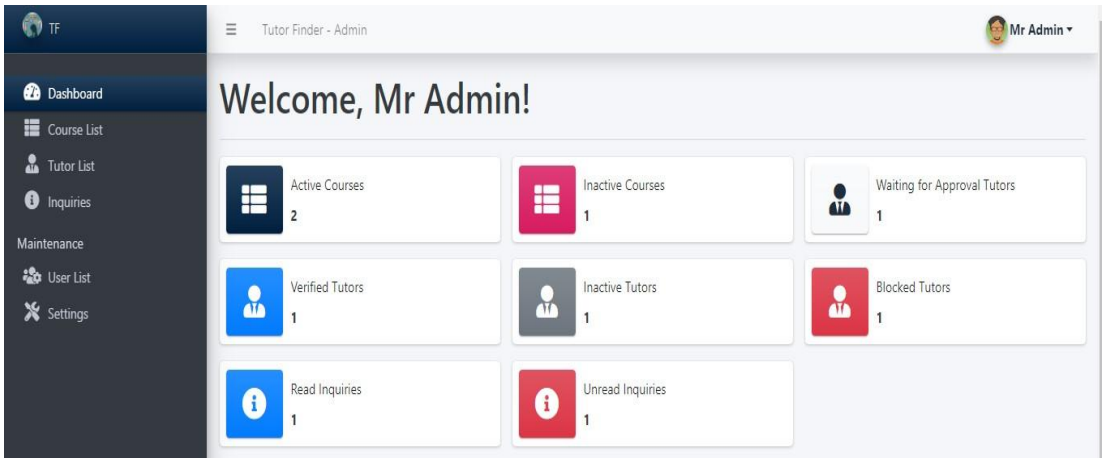
## 6.4 Admin Module

Admin has full control over the website and can manage the Tutor Finder.

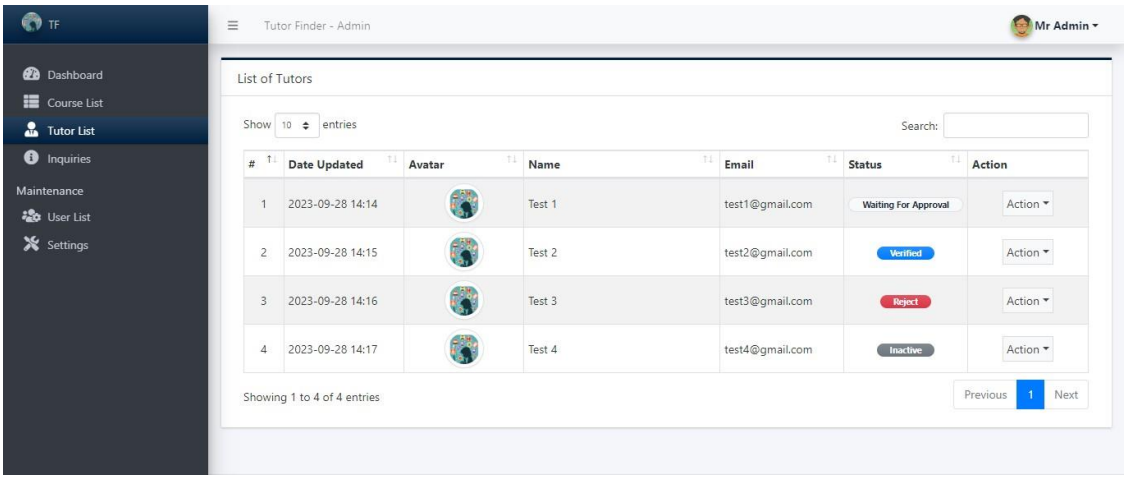
### 6.4.1 Login Page




### 6.4.2 Dashboard



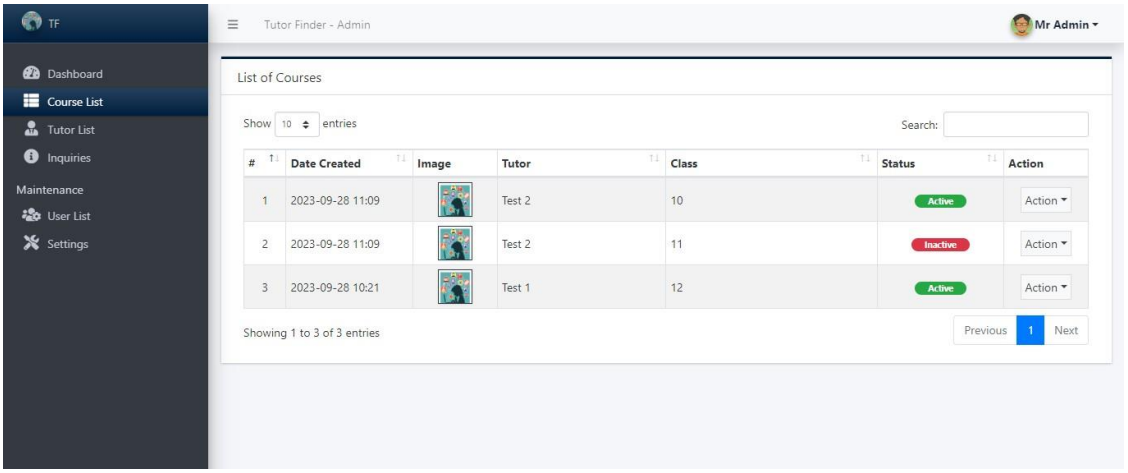
### 6.4.3 Tutor List



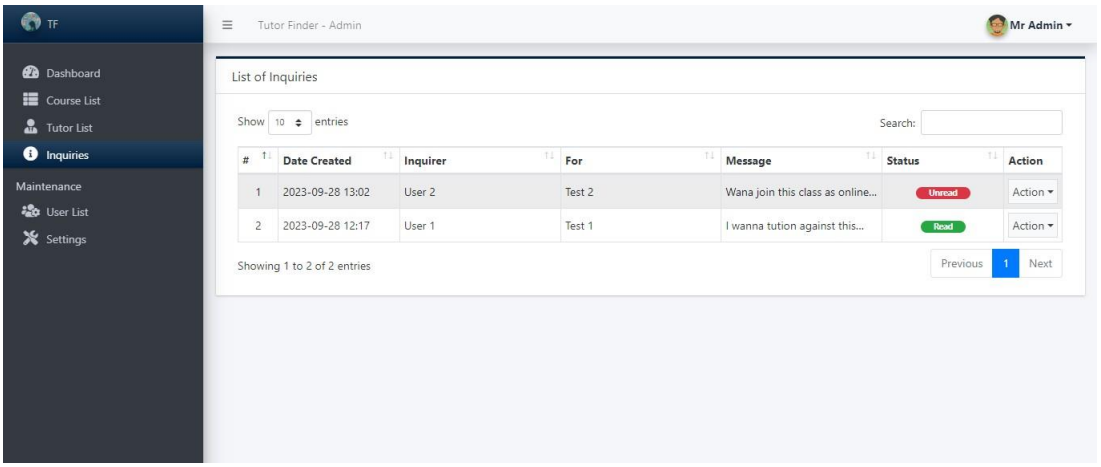

Here, admin can set status for tutors.

- **Waiting for Approval:** This status for those tutors, who will wait for any approval from admin.
- **Verified:** This status for those tutors, who will verified from admin.
- **Reject:** This status for those tutors, who will rejected from admin.
- **Inactive:** This status for those tutors, who will be blocked from admin.

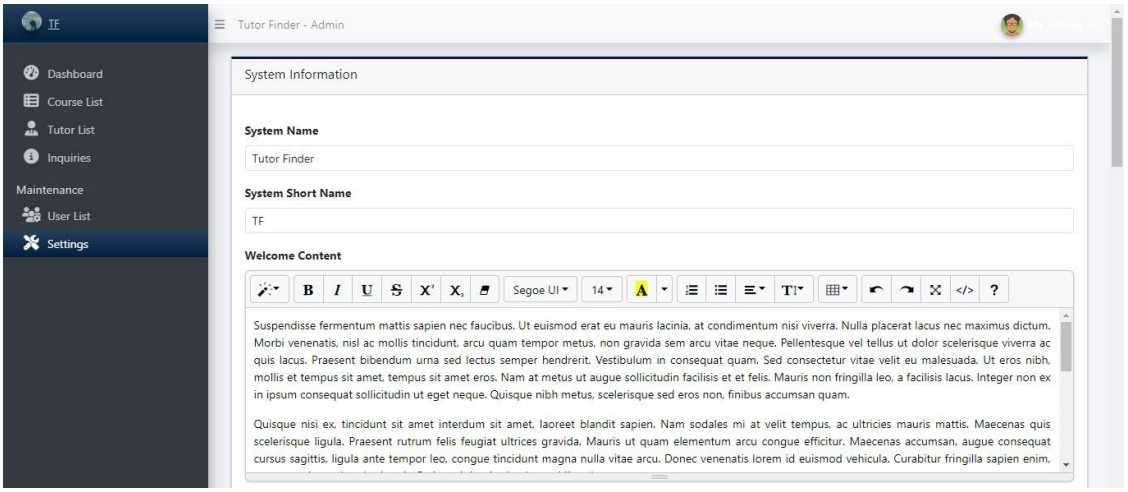
### 6.4.5 Course List

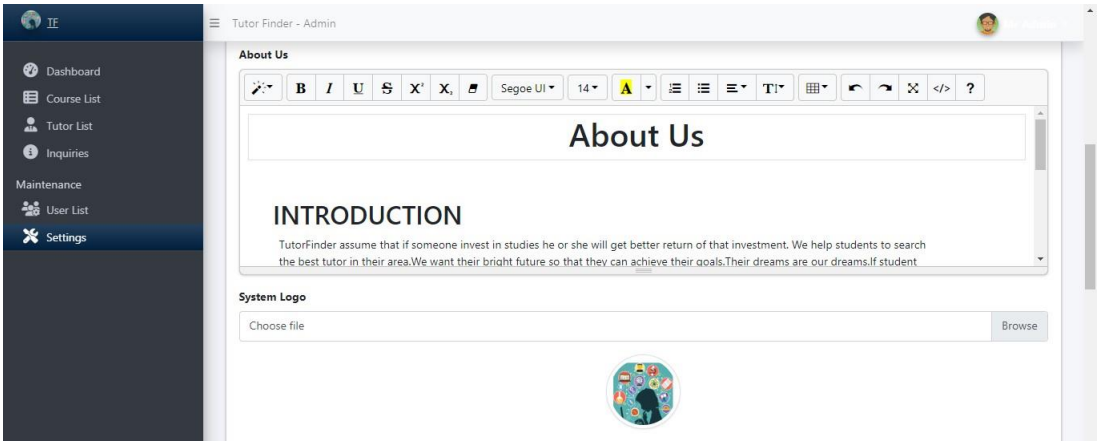


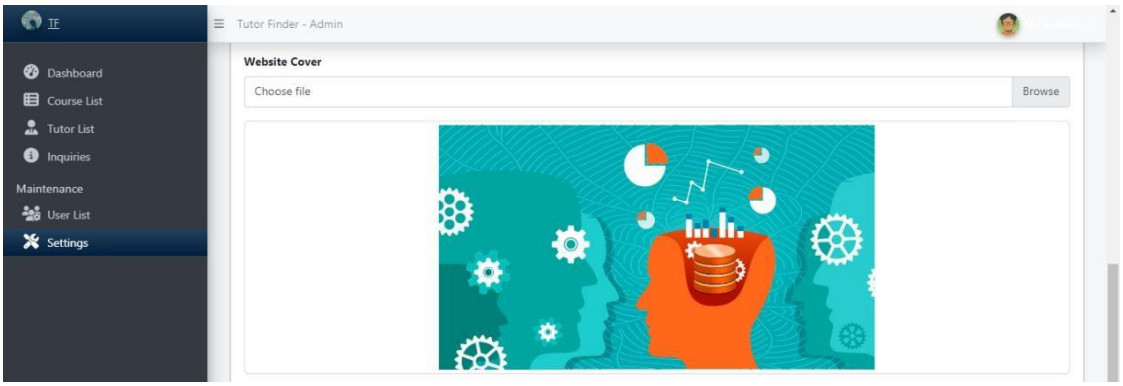

### 6.4.6 Inquiries

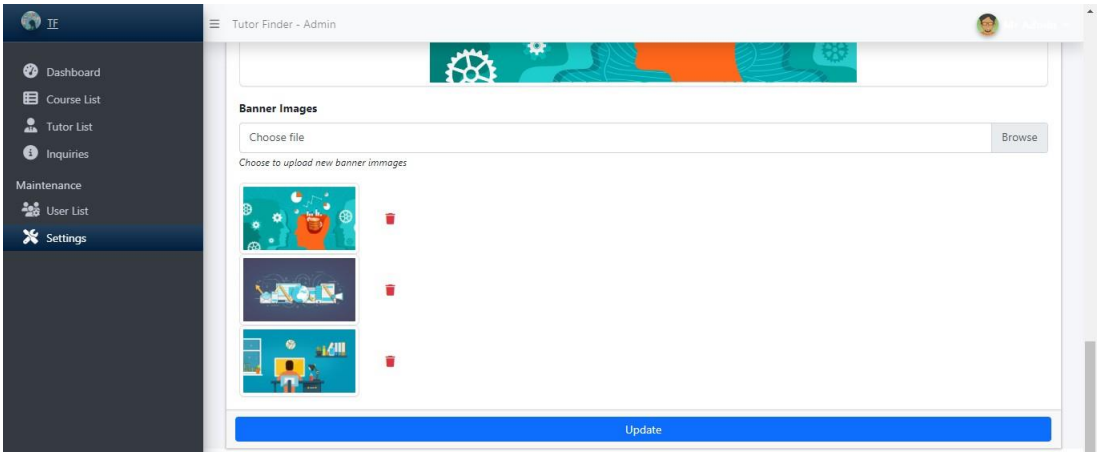


### 6.4.7 Settings




TF
Tutor Finder - Admin
Dashboard
Course List
Tutor List
Inquiries
Maintenance
User List
Settings
Website Cover
Choose file
Browse


TF
Tutor Finder - Admin
Dashboard
Course List
Tutor List
Inquiries
Maintenance
User List
Settings
Banner Images
Choose file
Browse
Choose to upload new banner immages
Update